

Remarks

This reply is responsive to the Office communication mailed November 1, 2005. Page and paragraph references are to that communication unless otherwise indicated.

Changes Made

Claim 1 has been amended to expressly recite an excluding step rather than just define it inferentially. Claim 1 has also been amended to recite that the static objects are embedded "in one or more defined storage areas" (page 6, line 7) in the dynamic file system and that the static data objects are excluded from actions performed on the dynamic file system "that would cause said data objects to occupy storage outside of said defined storage areas". Such actions would include, for example, changing the size of the static data object (page 7, line 25; page 8, line 3) or moving the static data object to a location outside the defined storage area (page 3, lines 8-9; page 11, lines 11-12). However, they would not include merely accessing a static data object (page 3, lines 14-15), deleting a static data object (page 7, lines 20-21), or updating a static data object where that updating does not involve moving the data object or changing its size (page 7, line 25).

New claim 13, dependent on claim 1, recites more particularly that the data objects are excluded from actions performed dynamically on the file system that would move the data objects outside of the defined storage areas (page 3, lines 8-9; page 11, lines 11-12). Similarly, new claim 14 recites more particularly that the data objects are excluded from actions performed dynamically on the file system that would enlarge the data objects beyond the defined storage areas (page 7, line 25; page 8, line 3). New claims 15 and 16 are similar to claims 13 and 14, but are dependent on program product claim 10. Finally, new claims 15 and 16 are similar to claims 13 and 14, but are dependent on apparatus claim 11.

Specification Objection

The Examiner objects to the specification as allegedly failing to provide antecedent basis for the claimed subject matter (page 2). More particularly, he alleges that the specification does not describe in clear and precise terms the manner and process of: (1) creating management actions; (2) excluding the static object from such management actions; and (3) making a kind of embedment in which one or more static data objects are embedded in the dynamic file system. Applicants respectfully traverse.

While the Examiner refers to applicants' "claimed subject matter", he quotes applicants' abstract, and the terms he questions come from the abstract rather than the claims. Thus, it is the abstract rather than the claims that speaks of "management actions" and a "kind of embedment". Since these terms do not even appear in the claims, the Examiner's objection is inapposite.

If instead the Examiner is requiring an antecedent basis for the steps of embedding one or more static data objects in the dynamic file system and excluding such data objects from actions performed dynamically on the file system, that is a legitimate inquiry, since these steps are recited in the claims. However, these claim recitations are clearly supported by the specification as originally filed, including the abstract.¹ Thus, the step of embedding one or more static data objects in the dynamic file system is described at page 3, lines 1-3 and 27-29 and page 4, lines 21-22, while the step of excluding the data objects from actions performed dynamically on the file system is described at page 3, lines 4-5, as well as page 11, lines 9-10 (the abstract).²

Accordingly, this objection is untenable and should therefore be withdrawn.

Claim Rejections — 35 U.S.C. § 112

Claims 1-7 and 10-12 stand rejected under 35 U.S.C. § 112, second paragraph, as being indefinite (page 3). More particularly, the Examiner contends that the metes and bounds of the

¹ The Examiner does not appear to be arguing here that these claimed steps are not enabled. He is merely arguing that the claim recitations lack antecedent support in the specification.

² The support for the newly added recitations is set forth in the section above ("Changes Made").

claim language are unclear because it is unclear "which operations, if indeed there are any actions at all, from which static objects are excluded" (page 3).

Claims 1-7 and 10-12 as amended, as well as new claims 13-18, are believed to avoid this ground for rejection, since they now clearly indicate the actions from which the static objects are excluded. The instances enumerated by the Examiner, in which actions are performed on static data objects, are consistent with the present claim language, since none of these actions causes the data objects to occupy storage outside of the defined storage areas, either by changing the size of the data objects or moving the data objects to a location outside the defined storage area.

Claim Rejection — 35 U.S.C. § 102

Claims 1-7 and 10-12 also stand rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent 6,220,510 to Everett et al. ("Everett") (page 4). Applicants respectfully traverse.

Everett describes a so-called multi-application IC card with a delegation feature. As shown in Fig. 1, the logical data space allocation 101 created for a particular application includes a static data space 103, a public data space 105, and a dynamic data space 107. The static data space 103 is non-volatile, whereas the public data space 105 and the dynamic data space 107 are volatile (col. 4, lines 53-58). The public data space 105 and the dynamic data space 107 differ from each other in that the public data space 105 is shared with other applications, whereas the dynamic data space is used by only a single application (col. 2, lines 44-48). The term "dynamic" is thus somewhat of a misnomer; a better term might be "private".

In reading claim 1 onto Everett, the Examiner equates applicants' static objects with Everett's static data space 103, applicants' dynamic file system with Everett's "dynamic" data space 107, and applicants' dynamically performed actions with various actions on the static data space described in patent. This comparison, however, is misplaced.

In the first place, Everett's dynamic data space 107 is not a "file system" as claimed by applicants. As applicants have noted previously, the online encyclopedia Webopedia³ defines a "file system" as "[t]he system that an operating system or program uses to organize and keep track of files". Similarly, the online encyclopedia Wikipedia defines a file system as "a method for storing and organizing computer files and the data they contain to make it easy to find and access them".⁴ A "file system" is thus a programming construct and not simply anything (hardware, software or whatever) that might contain a file. It certainly does not include Everett's dynamic data space 107, which is more properly characterized as a segment of a virtual address space (col. 4, lines 24-31). Indeed, the only time the term "file" even appears in the specification is in the variable name `selected_file_application_id` in steps 301 and 301 in Fig. 3 and steps 407 and 411 in Fig. 4.

Even assuming, for the sake of argument, that Everett's dynamic data space 107 constitutes a "file system", the static data space 103 is not "embedded" in that structure as claimed by applicants. As defined by applicants in the specification, the term "embedding" here means either that the static data object is surrounded on both sides by the dynamic file system or that the static data object lies entirely within the dynamic file system and is coterminous with one end thereof (page 3, lines 27-29). Everett's static data space 103 is neither of these things. Not only is it not embedded in the dynamic data space 107, but it is not even contiguous to it, being separated from the dynamic data space by the public data space 105. Even if dynamic data space 107 were contiguous to the static data space 103, as the public data space 105 appears to be in Fig. 2, the static data space would not be "embedded" in it, since they would still be disjoint. They would of necessity always be disjoint, since the static data space 103 corresponds to a non-volatile memory area, while the public data space 105 and the dynamic data space 107 map to volatile memory areas.

Accordingly, not only are claims 1-7 and 10-12 not anticipated by Everett, but they clearly distinguish patentably over that reference, so the Examiner's rejection of these claims on the reference is untenable.

³ Found at <http://www.webopedia.com/>.

⁴ This definition may be found at http://en.wikipedia.org/wiki/File_system.

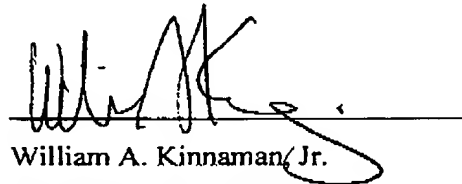
Conclusion

For the foregoing reasons, claims 1-7 and 10-12 as amended and new claims 13-18 comply with 35 U.S.C. § 112 and distinguish patentably over the reference cited. Accordingly, applicants respectfully request that the outstanding rejection be withdrawn.

Respectfully submitted,

E.-M. HAMANN et al.

By



William A. Kinnaman, Jr.

Registration No. 27,650

Phone: (845) 433-1175

Fax: (845) 432-9601

WAK/wak